

SYMMETRIC PRIMAL-DUAL SIMPLEX ALGORITHM FOR LINEAR PROGRAMMING

Keshava Prasad Halemane
Department of Mathematical and Computational Sciences,
Karnataka Regional Engineering College Surathkal,
Post Srinivasnagar, Mangalore-575 025,
Karnataka State, India.

ABSTRACT

A novel iterative algorithm to solve linear programming (LP) problems based upon a newly identified set of four (and/or possibly six) distinct types of simplex pivoting steps defined over a symmetric primal-dual pair of LP, is presented. The two (or possibly four) types of classical (standard) simplex pivots are the Primal Standard Pivot with positive (and/or zero) indicator, and the Dual Standard Pivot with negative (and/or zero) indicator. The newly identified pivot types are: the Primal Tricky Pivot with positive indicator and the Dual Tricky Pivot with negative indicator.

The Symmetric Primal-Dual Simplex Algorithm is an iterative solution strategy, wherein each iteration consists of the selection and application of exactly one pivoting step, chosen in a well defined systematic preference order (for our convenience, at least), from among the possibly four (or even six) distinct types of pivots, that may be valid depending on the Tableaux data entries.

For expressional convenience, this algorithm is discussed using the Tucker's Compact Symmetric Tableaux (CST) for Linear Programming problems expressed in the standard (canonical) form representing the symmetric Primal-Dual Pair. An analysis of the evolution of the Tableaux data entries pattern observed as iterations proceed, is also presented. A classification of the LP problem into one of the six types is identified based on the data entries pattern observed in the final Tableaux.

1. INTRODUCTION:

Linear Programming (LP) problem represents one of the most widely used class of numerical/quantitative computational models, for which any possible improved solution technique would certainly be highly desirable. Of course, there has been several alternative solution strategies suggested including the classical simplex method of Dantzig [1] and several variations thereof, followed by recent polynomial time algorithms, namely the Ellipsoid Method of Khachiyan [2, 3] and the Karmarkar Algorithm [4] both classified now as belonging to Interior Point Algorithms.

In this paper, a novel generalization of the classical Simplex Method of Dantzig is presented. The fundamental concept in Dantzig's Simplex Method consists of moving from one basis (Tableaux) to a neighbouring basis (Tableaux), by a single exchange between a non-basic (entering) variable and a basic (leaving) variable, appropriately selected. We associate the term "Simplest Algorithm" for our method because this fundamental feature is being preserved. The proposed algorithm is an enhancement over the Dantzig's Simplex Method, in terms of providing a wider scope for the selection of the pivots, as can be seen in the ensuing discussions. The development of the algorithm is mainly supported by the identification of a set of possibly four (or even six) distinct types of simplex pivot selections, maintaining a sense of symmetry between the primal and the dual problems. Also, for expressional efficiency or at least convenience, the solution strategy and the analysis thereof are all based on the Tucker's Compact Simplex Tableaux for LP expressed in the standard (canonical) form representing the symmetric primal-dual pair. Hence, we justify the chosen name for the present algorithm: "Symmetric Primal-Dual Simplex Algorithm for Linear Programming".

2. PRELIMINARIES:

We will go through some well known preliminaries for the sake of establishing the notational conventions used in this paper. The Symmetric Primal-Dual Pair of LP in the Standard Canonical Form is as follows.

Primal Problem:

$$\begin{array}{ll} \text{Maximize} & c.x = f \\ \text{s.t.} & A.x \leq b \\ & x \geq 0 \end{array} \quad (1)$$

Dual Problem:

$$\begin{array}{ll} \text{Minimize} & v.b = g \\ \text{s.t.} & v.A \geq c \\ & v \geq 0 \end{array} \quad (2)$$

The matrix-dimensional descriptions for each of the problem parameters/data in (1) and (2) above are as follows:

x	Primal decision Variables	nx1 vector
c	Primal objective function coeffs	1xn vector
f	Primal objective function value	1x1 scalar
A	Primal constraint coeffs matrix	mxn matrix
b	Primal constraint upper bound	mx1 vector
v	Dual decision variables	1xm vector
g	Dual objective function value	1x1 scalar

We introduce the $m \times 1$ vector y of slack variable to (1) and the $1 \times n$ vector u of surplus variables to (2) to rewrite this Symmetric Primal-Dual pair in an equivalent form as follows :

Primal Problem:

$$\begin{aligned} \text{Maximize} \quad & c.x + 0.y = f \\ \text{s.t.} \quad & A.x + I_m.y = b \\ & x, y \geq 0 \end{aligned} \quad (3)$$

Dual Problem:

$$\begin{aligned} \text{Minimize} \quad & v.b + u.0 = g \\ \text{s.t.} \quad & v.A - u.I_n = c \\ & v, u \geq 0 \end{aligned} \quad (4)$$

This Symmetric Primal-Dual pair is represented in the Compact Symmetric Tableaux (CST) as:

$$\begin{array}{cc|c} & x_j & -1 \\ v_i & a_{ij} & b_i \\ \hline -1 & c_j & 0 \end{array} \quad \begin{array}{l} = -y_i \\ = f \end{array} \quad (5)$$

$\parallel \qquad \parallel$
 $u_j \qquad g$

For the LP problem pair (1) & (2) or equivalently (3) & (4) the Tableaux (5) represents the initial Tableaux indicating the initial basic solution (IBS) wherein y_i are the primal basic variables in the initial basis associated (one to one permanent association) with v_i the dual non-basic variables, and x_j are the primal non-basic variables associated (one to one permanent association) with u_j the dual basic variables, in the initial basis. Note that x_j (and a -1) are column-labels and v_i (and a -1) are row labels in the Tableaux (5), and the way to interpret (read) the Tableaux is as follows:

Primal Problem:

$$\sum_{j \in C} a_{ij} \cdot x_j - b_i = -y_i, \quad i \in R \text{ (row index)} \quad (6)$$

$$\sum_{j \in C} c_j \cdot x_j - 0 = f \quad \text{(function to be maximized)}$$

Dual Problem:

$$\sum_{i \in R} v_i \cdot a_{ij} - c_j = u_j, \quad j \in C \quad (\text{Column index}) \quad (7)$$

$$\sum_{i \in R} v_i \cdot b_i - 0 = g \quad (\text{function to be minimized})$$

wherein the variables x_j, y_i, v_i, u_j are all considered to be non-negative.

3. ALGEBRA (AIRTHMETIC) OF THE PIVOTING PROCESS:

It is an interesting but simple enough exercise to observe that once a pivot element is selected, the actual pivoting process (the algebra and hence the arithmetic operations) is the same irrespective of the pivot selection; for example whether it is a primal pivot or a dual pivot. Hence it suffices to present here a single (common) set of operations representing that pivoting process.

For the sake of generality, let us imagine that we are somewhere in the middle of solving a LP problem, and have the system model represented by a Tableaux as follows:

$$\begin{array}{cc|c} & z_j^N & -1 \\ w_i^N & \alpha_{ij} & \beta_i \\ -1 & \gamma_j & \delta \end{array} \quad \begin{array}{l} = -z_i^B \\ = f \end{array} \quad (8)$$

$\parallel$ $\parallel$
 w_j^B g

By the nature of the sequence of elementary row (column) operation being performed during any pivoting process, the system model represented by (8) is equivalent to that represented by the initial Tableaux (5) which corresponds to the primal-dual pair (6) & (7). The transformed version of the primal-dual pair directly expressed by (8) is as follows.

Primal: (9)

$$z_i^B = \beta_i - \sum_{j \in C} \alpha_{ij} \cdot z_j^N, \quad i \in R \text{ (row index)}$$

$$f = -\delta + \sum_{j \in C} \gamma_j \cdot z_j^N, \quad \text{(function to be maximized)}$$

Dual: (10)

$$w_j^B = -\gamma_j + \sum_{i \in R} w_i^N \cdot \alpha_{ij}, \quad j \in C \text{ (column index)}$$

$$g = -\delta + \sum_{i \in R} w_i^N \cdot \beta_i \quad \text{(function to be minimized)}$$

The effect of a pivoting operation on (9) & (10) performed with a chosen pivot element α_{IJ} is exactly to affect an exchange between the variables indicated by I and J in (9) and (10). That is, z_j^N is entered into primal basis in exchange for z_i^B in (9), and w_i^N is entered into dual basis in exchange for w_j^B in (10). Suppose we have chosen the pivot element α_{IJ} using some appropriate pivot selection scheme, and we would like to derive the resulting Tableaux. Let the resulting Tableaux be indicated as:

$$\begin{array}{cc|c} & (z_j^N)' & -1 \\ (w_i^N)' & (\alpha_{ij})' & (\beta_i)' \\ \hline -1 & (\gamma_j)' & (\delta)' \end{array} = \begin{array}{l} -(z_i^B)' \\ (f)' \end{array} \quad (11)$$

$\parallel \qquad \qquad \parallel$
 $(w_j^B)' \qquad \qquad (g)'$

Then the process of deriving (11) from (8) constitute the algebra (arithmetic) of the pivoting process and is detailed below (without any derivational justifications):

$$\begin{array}{lll} (\alpha_{IJ})' \leftarrow (1/\alpha_{IJ}); & (\alpha_{Ij})' \leftarrow (\alpha_{Ij})/\alpha_{IJ}; & (\beta_I)' \leftarrow (\beta_I)/\alpha_{IJ}; \\ (\alpha_{iJ})' \leftarrow -(\alpha_{iJ})/\alpha_{IJ}; & & (\gamma_J)' \leftarrow -(\gamma_J)/\alpha_{IJ}; \\ (\alpha_{ij})' \leftarrow \alpha_{ij} - (\alpha_{iJ}/\alpha_{IJ})\alpha_{IJ}; & & (\beta_i)' \leftarrow \beta_i - (\beta_I/\alpha_{IJ})\alpha_{IJ}; \\ (\gamma_j)' \leftarrow \gamma_j - (\alpha_{iJ}/\alpha_{IJ})\gamma_J; & & (\delta)' \leftarrow \delta - (\beta_I/\alpha_{IJ})\gamma_J; \end{array}$$

followed by an exchange of the labels – associated with the row I and the column J; that is effectively:

$$\begin{aligned}
 & (z_j^N)' \leftarrow z_i^B; & (z_i^B)' \leftarrow z_j^N; \\
 & (z_j^N)' \leftarrow z_j^N; & (z_i^B)' \leftarrow z_i^B; \\
 \text{and} \\
 & (w_j^B)' \leftarrow w_i^N; & (w_i^N)' \leftarrow w_j^B \\
 & (w_j^B)' \leftarrow w_j^B; & (w_i^N)' \leftarrow w_i^N; \\
 & \text{for } i \in R \setminus \{I\} \text{ and } j \in C \setminus \{J\}.
 \end{aligned}$$

This completes the pivoting process.

4. SIMPLEX PIVOT SELECTION SCHEMES:

There are four (two pairs) fundamental types of pivot selection schemes namely Primal Standard Pivot (PSP), Dual Standard Pivot (DSP), Primal Tricky Pivot (PTP) and Dual Tricky Pivot (DTP). Further a primal/dual standard pivot can again be classified into one with positive/negative indicator (that is PSPPI and DSPNI) and one with zero indicator (that is PSPZI and DSPZI), thus resulting in a set of possible six distinct types of pivot selection schemes, that are available for simplex pivoting process in solving linear programming problems. The algebra of the pivot selection schemes are given below in Figure-1, along with a schematic representation of the Tableaux data entries pattern that leads to such pivot selection.

It can be easily observed that a primal (dual) tricky pivot with zero indicator is essentially the same as dual (primal) standard pivot with negative (positive) indicator, wherein the resultant minimum-ratio comes out to be zero.

5. EFFECT OF PIVOTING OPERATION:

It is useful at this point to make a few observations regarding the effect of pivoting operation, in each of the above pivot selection schemes.

DSPNI brings about an immediate improvement in the primal feasibility w.r.t. the pivot row, without deterioration of dual feasibility. The extent of this improvement in primal feasibility can be measured by the corresponding improvement (decrease) observed in the value of the dual objective function, given by $|\beta_I \gamma_J / \alpha_{IJ}|$.

PSPPI brings about an immediate improvement in the dual feasibility w.r.t. the pivot column, without deterioration of primal feasibility. The extent of this improvement in dual feasibility can be measured by the corresponding improvement (increase) observed in the value of the primal objective function, given by $|\beta_I \gamma_J / \alpha_{IJ}|$.

4.1 Dual Standard Pivot with Negative Indicator, DSPNI:

•	—	—	←
•	•	⊕	
+	Θ	•	

↓

$$I \in \{i \in R \mid \beta_i < 0\};$$

$$J \leftarrow \arg \min_{j \in C} \{(\gamma_j / \alpha_{ij}) \mid \beta_i < 0; \gamma_j \leq 0; \alpha_{ij} < 0\};$$

4.2 Primal Standard Pivot with Positive Indicator, PSPPI:

•	•	—	
+	•	⊕	→
+	Θ	•	

↑

$$J \in \{j \in C \mid \gamma_j > 0\};$$

$$I \leftarrow \arg \min_{i \in R} \{(\beta_i / \alpha_{ij}) \mid \beta_i \geq 0; \gamma_j > 0; \alpha_{ij} > 0\};$$

4.3 Primal Tricky Pivot (with Positive Indicator), PTPPI:

—	•	—	
•	•	⊕	→
+	Θ	•	

↑

$$J \in \{j \in C \mid \gamma_j > 0\};$$

$$I \leftarrow \arg \max_{i \in R} \{(\beta_i / \alpha_{ij}) \mid \beta_i < 0; \gamma_j > 0; \alpha_{ij} < 0\};$$

4.4 Dual Tricky Pivot (with Negative Indicator), DTPNI:

+	•	—	←
•	•	⊕	
+	Θ	•	

↓

$$I \in \{i \in R \mid \beta_i < 0\};$$

$$J \leftarrow \arg \max_{j \in C} \{(\gamma_j / \alpha_{ij}) \mid \beta_i < 0; \gamma_j > 0; \alpha_{ij} > 0\};$$

4.5 Dual Standard Pivot with Zero Indicator, DSPZI:

•	•	—	
•	—	0	←
•	•	+	
+	Θ	•	

↓

$$I \in \{i \in R \mid \beta_i = 0\};$$

$$J \leftarrow \arg \min_{j \in C} \{(\gamma_j / \alpha_{ij}) \mid \beta_i = 0; \gamma_j \leq 0; \alpha_{ij} < 0\};$$

4.6 Primal Standard Pivot with Zero Indicator, PSPZI:

•	•	•	—	
•	+	•	⊕	→
+	0	—	•	

↑

$$J \in \{j \in C \mid \gamma_j = 0\};$$

$$I \leftarrow \arg \min_{i \in R} \{(\beta_i / \alpha_{ij}) \mid \beta_i \geq 0; \gamma_j = 0; \alpha_{ij} > 0\};$$

Figure-1. Six types of Simplex Pivot Selection Schemes

— negative; 0 zero; + positive; Θ non-positive; * any value; ⊕ non-negative; • un-analyzed

PTPPI brings about an immediate improvement in the primal feasibility w.r.t. the pivot row, without any concern to the dual feasibility. The extent of this improvement in primal feasibility can be measured by the corresponding improvement (increase) in the value of the primal objective function, given by $|\beta_I \gamma_J / \alpha_{IJ}|$.

DTPNI brings about an immediate improvement in the dual feasibility (w.r.t. the pivot column, at least) without any concern to the primal feasibility. The extent of this improvement in dual feasibility can be measured by the corresponding improvement (decrease) in the value of the dual objective function, given by $|\beta_I \gamma_J / \alpha_{IJ}|$.

The other two types of pivot selections namely PSPZI and DSPZI are utilized to affect a change in the basis, in situations with multiple-solutions and/or degeneracy, and do not in any way affect the primal or dual feasibilities. However, they certainly play an important role in the overall solution strategy.

From the above, one can observe that as a general, uniform and common basis for a local effectiveness measure (lem) of a particular pivoting operation (applicable for any and every iteration, for both primal and dual) we can use the absolute value of the change in the value of the objective function, given by $lem(I, J) = \text{abs}(\beta_I \gamma_J / \alpha_{IJ})$. Although it is not specifically suggested here, one can opt to choose a pivot possibly to maximize this local effectiveness measure (lem) in every iteration among the possible pivots of a particular type or even among of all the possible pivots of all the possible types. Even, if done so, it cannot be guaranteed (as per worst case analysis) to minimize the overall number of iterations required for reaching an optimum solution. It requires further research work to thoroughly understand, analyze and incorporate the concept of a possible “local effectiveness measure (lem)” for a single simplex pivoting operation to the fullest extent, that would be equivalent to a guaranteed improvement in some corresponding global effectiveness measure (gem) defined appropriately for the given LP problem; in developing an efficient solution strategy. For now, let us come to the main algorithm itself.

6. SYMMETRIC PRIMAL-DUAL SIMPLEX ALGORITHM:

The six distinct types of pivot selection schemes are considered in the following order of preference:

$$\{\{ \text{DSPNI}, \text{PSPPI} \}, \{ \text{PTPPI}, \text{DTPNI} \}, \{ \text{DSPZI}, \text{PSPZI} \} \}$$

At every iteration, an attempt is made to select a pivot element, by checking the possible pivot selections belonging to one of the above six types of pivot selection schemes in the preference order pre-specified.

That is, a pivoting operation is executed with a pivot element of the type -

- i. DSPNI if possible,
- ii. PSPPI only if none of DSPNI is possible,
- iii. PTPPI only if none of DSPNI nor PSPPI is possible,
- iv. DTPNI only if none of DSPNI, PSPPI, PTPPI is possible,

and when none of the above four is possible, go for

- v. DSPZI and/or
- vi. PSPZI in order to explore the possible alternative basic solutions.

It is to be noted that depending upon the actual data entries in the Tableaux, a pivot selection of specific type which was not possible in an earlier iteration, can become possible in a later iteration, sometimes even in the very next following iteration. That is why it is a crucial part of the algorithm to check in each and every iteration, for each of the six types of possible pivot selection schemes in the pre-specified order.

7. EVOLUTION OF TABLEAUX DATA ENTRIES PATTERN:

As a means to show the convergence of the algorithm to one of the possible six termination patterns (discussed later), an analysis of the evolution of the Tableaux data entries pattern is given here. Figure-2 gives the six different patterns applicable when each of the six different pivot selection schemes may be chosen as a possibility in some iteration during the process of solving a LP problem.

Figure-2 needs to be read keeping in mind the algebra of the pivoting process and the details regarding the nature of the pivot selection itself. To facilitate the analysis of the possible termination patterns, it is useful to look at a systematic classification of all the 27 possible combinations of Tableaux data entries pattern as shown in Figure-3. Here, the possible situations that would lead to a successful pivoting are indicated- they add up to ten, including the four situation corresponding to multiplicity/degeneracy. Others wherefor no pivot selection is possible, are marked according to the nature of the termination, by indicating F for basic-feasible-finite, Φ for infeasible and ∞ for non-basic-feasible-infinite, corresponding to both primal and dual variables. Note that this classification is as per the quite well established approach, although a refinement is indicated to distinguish the case wherein primal (dual) has an infinite non-basic optimum with finite value for the objective function while the dual (primal) has a feasible-finite optimum. This distinction from the classical approach arises because we give primary emphasis on the classification based on the nature of the decision variables at termination, and give secondary emphasis on the finiteness (or otherwise) of the objective function value.

(a) DSPNI

*	-	-	←
*	*	⊕	
+	Θ	*	
	↓		

(b) PSPPI

*	⊕	-	
+	*	⊕	→
+	Θ	*	
	↑		

(c) PTPPI

-	⊕	-	→
Θ	*	⊕	
+	Θ	*	
	↑		

(d) DTPNI

+	⊕	-	←
Θ	*	⊕	
+	Θ	*	
	↓		

(e) DSPZI

0	⊕	⊕	-	
Θ	-	-	0	←
Θ	*	*	+	
+	0	-	*	
	↓	↓		

(f) PSPZI

0	⊕	⊕	-	
Θ	+	*	0	→
Θ	+	*	+	→
+	0	-	*	
	↑			

Figur-2. Evolution of Tableaux data entries pattern

— negative; 0 zero; + positive;
 Θ non-positive; * any value; ⊕ non-negative

$\begin{array}{ c c } \hline 0 & - \\ \hline - & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline 0 & 0 \\ \hline - & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline 0 & + \\ \hline - & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline + & - \\ \hline - & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline + & 0 \\ \hline - & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline + & + \\ \hline - & * \\ \hline \end{array} F$
∞	F	F	∞	∞	F
$\begin{array}{ c c } \hline 0 & - \\ \hline 0 & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline 0 & 0 \\ \hline 0 & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline 0 & + \\ \hline 0 & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline + & - \\ \hline 0 & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline + & 0 \\ \hline 0 & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline + & + \\ \hline 0 & * \\ \hline \end{array} F$
∞	F	F	∞	F	F
$\begin{array}{ c c } \hline 0 & - \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline 0 & 0 \\ \hline + & * \\ \hline \end{array} \infty$	$\begin{array}{ c c } \hline 0 & + \\ \hline + & * \\ \hline \end{array} \infty$	$\begin{array}{ c c } \hline + & - \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline + & 0 \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline + & + \\ \hline + & * \\ \hline \end{array} \Phi$
Φ	Φ	Φ	Φ	Φ	Φ
$\begin{array}{ c c } \hline - & - \\ \hline - & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline - & 0 \\ \hline - & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline - & + \\ \hline - & * \\ \hline \end{array} F$			
DSPNI	DSPZI				
$\begin{array}{ c c } \hline - & - \\ \hline 0 & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline - & 0 \\ \hline 0 & * \\ \hline \end{array} F$	$\begin{array}{ c c } \hline - & + \\ \hline 0 & * \\ \hline \end{array} \infty$			
DSPNI	DSPZI				
$\begin{array}{ c c } \hline - & - \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline - & 0 \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline - & + \\ \hline + & * \\ \hline \end{array} \Phi$			
PTPPI					
$\begin{array}{ c c } \hline - & - \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline - & 0 \\ \hline + & * \\ \hline \end{array} \Phi$	$\begin{array}{ c c } \hline - & + \\ \hline + & * \\ \hline \end{array} \Phi$			
Φ	Φ	Φ			

Figure-3. Twenty seven distinct combinations of Tableaux data entries pattern.

— negative; 0 Zero; + positive; * any value

Thus the seventeen (out of the total of twenty seven) situations that lead to non pivoting termination can be classified into six distinct classes as follows

- i) Six situations with $P = F, D = F$
- ii) One situation with $P = F, D = \infty$
- iii) One situation with $P = \infty, D = F$
- iv) Four situations with $P = \infty, D = \Phi$
- v) Four situations with $P = \Phi, D = \infty$
- vi) One situation with $P = \Phi, D = \Phi$

Four among the remaining ten (out of the total of twenty seven) situations that indicate pivoting possibilities can be included in case (i) above and treated as possible termination situations because further pivoting would be useful only to the extent of discovering multiple optima for the primal/dual. The other six correspond to the six types of pivot selection schemes indicated. Figure-4 summarises these observations.

From the above discussion and the observations presented in Figures 1, 2, 3 and 4, it is clear that the algorithm converges to one of the six distinct classes of termination situations and the given LP-Problem can be classified accordingly.

8. SOME GENERAL COMMENTS:

It is worth noting here that using the above proposed algorithm one can solve any LP problem, by first converting it into the standard/canonical form and then applying the proposed algorithm. In performing such transformation, it is possible to enhance the overall efficiency by the following approach:

- i) Free variables can be replaced by non-negative variables not double in number, but only just one extra in number.
- ii) Equations can be replaced by inequalities not double in number, but only just one extra in number
- iii) No need for use of artificial variables.
- iv) The initial basic solution need not necessarily be feasible.

Degeneracy and possible cycling can be handled in the usual method, by lexicographic ordering of the variables (equivalent and in fact supported by the concept of polynomial perturbations technique).

Also it is possible to have alternative ordering of the pivot selection schemes incorporated in the algorithm within each of the three subsets indicated in section 6 above, although the relative ordering of these three subsets themselves is not to be altered.

Although rigorous mathematical analysis of the computational complexity of this algorithm is yet to be studied in detail, it seems very likely to perform at least as good as (if not better than) the existing solution techniques for linear programming.

	D = F	D = ∞	D = Φ																		
P = F	<table><tr><td>*</td><td>Θ</td><td>0</td></tr><tr><td>⊕</td><td>*</td><td>+</td></tr><tr><td>0</td><td>−</td><td>•</td></tr></table> 10	*	Θ	0	⊕	*	+	0	−	•	<table><tr><td>+</td><td>+</td><td>0</td></tr><tr><td>⊕</td><td>*</td><td>+</td></tr><tr><td>0</td><td>−</td><td>•</td></tr></table> 1	+	+	0	⊕	*	+	0	−	•	
*	Θ	0																			
⊕	*	+																			
0	−	•																			
+	+	0																			
⊕	*	+																			
0	−	•																			
P = ∞	<table><tr><td>−</td><td>Θ</td><td>0</td></tr><tr><td>−</td><td>*</td><td>+</td></tr><tr><td>0</td><td>−</td><td>•</td></tr></table> 1	−	Θ	0	−	*	+	0	−	•		<table><tr><td>Θ</td><td>*</td><td>⊕</td></tr><tr><td>+</td><td>Θ</td><td>•</td></tr></table> 4	Θ	*	⊕	+	Θ	•			
−	Θ	0																			
−	*	+																			
0	−	•																			
Θ	*	⊕																			
+	Θ	•																			
P = Φ		<table><tr><td>⊕</td><td>−</td></tr><tr><td>*</td><td>⊕</td></tr><tr><td>Θ</td><td>•</td></tr></table> 4	⊕	−	*	⊕	Θ	•	<table><tr><td>0</td><td>⊕</td><td>−</td></tr><tr><td>Θ</td><td>*</td><td>⊕</td></tr><tr><td>+</td><td>Θ</td><td>•</td></tr></table> 1	0	⊕	−	Θ	*	⊕	+	Θ	•			
⊕	−																				
*	⊕																				
Θ	•																				
0	⊕	−																			
Θ	*	⊕																			
+	Θ	•																			

Figure-4. Six categories for Tableaux data entries pattern at termination.

- negative; 0 zero; + positive;
 Θ non-positive; * any value; $\oplus$ non-negative; • un-analyzed

REFERENCES:

1. Dantzig, G. B. "Linear Programming and Extensions" Princeton University Press, 1963.
2. Khachiyan L. G. "A Polynomial algorithm in linear programming" (in Russian) Doklady Akademi i Nauk SSSR Vol. 244 (1979) pp 1093 – 1096.
[English Translation: Soviet Mathematics Doklady, Vol. 20 (1979) pp 191 – 194].
3. Khachiyan L. G. "Polynomial Algorithms in Linear Programming" (in Russian) Zhurnal Vychislitelnoi Matematiki i Matematicheskai Fiziki, vol. 20 (1980) pp 51-68.
[English Translation: USSR Computational Mathematics and Mathematical Physics, Vol. 20 (1980) pp 53-72]
4. Karmarkar N. "A new polynomial Algorithm for linear programming" Combinatorica, Vol. 4 (1984) pp 373-395.